DOCUMENT RESUME

ED 160 054 IR 006 117

| | |
|---|---|
| AUTHOR | Rouse, Sandra H.; Rouse, William B. |
| TITLE | Design of a Model-Based Online Management Information System for Interlibrary Loan Networks. Project Report No. 7. |
| INSTITUTION | Illinois Univ., Urbana. Coordinated Science Lab. |
| SPONS AGENCY | Illinois State Library, Springfield. |
| PUB DATE | Dec 77 |
| NOTE | 43p.; One of a series entitled: A Mathematical Model of the Illinois Interlibrary Loan Network; For related documents, see ED 101 667, ED 107 287, ED 124 179, ED 126 879, ED 134 201 |
| EDRS PRICE | MF-$0.83 HC-$2.06 Plus Postage. |
| DESCRIPTORS | Data Collection; Decision Making; *Information Processing; *Interlibrary Loans; *Library Administration; *Library Networks; Library Research; *Management Information Systems; *Mathematical Models |
| IDENTIFIERS | ILLINET; *Illinois Library and Information Network |

ABSTRACT

The inputs to mathematical models of library operations require periodic updating. The data collection effort necessary for such updating can be quite expensive; however, if the data collected can serve multiple purposes, the costs can perhaps be justified. A management information system (MIS) offers a vehicle for multiple uses of data, and thus offers an appropriate environment for implementation of models of library operations. The design of a model-based MIS is discussed in terms of three types of variables--mathematical/statistical, information processing, and human factors. General issues in these three areas are considered first, followed by a discussion of how these issues have been specifically addressed in the context of a prototype design of an online management information system for an interlibrary loan network. Considered in the discussion are MIS control, data base design, data entry, manager-MIS interaction, and MIS operating costs. Experience in modeling networks and in performing the work on which this report is based has led the authors to conclude that mathematical models can be successfully implemented. However, the question remains of whether or not a particular quantitative measure of work can be assigned to management information. The prototype model is currently operating at the University of Illinois on a DEC System-10. (Author/JPF)

**********
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
**********

# Design of a Model-Based Online Management Information System

for

# Interlibrary Loan Networks

ED160054

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

Project Report No. 7 In A Series Entitled:

A Mathematical Model Of The Illinois Interlibrary Loan Network

Submitted to

Illinois State Library

Sandra H. Rouse

William B. Rouse

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

William B. Rouse

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

Coordinated Science Laboratory

University of Illinois at Urbana-Champaign

Urbana, Illinois 61801

December 1977

This research was made possible by a grant from the Illinois State Library under the Illinois Program for Title I of the Federal Library Services and Construction Act.

IR006117

# FOREWORD

The primary purpose of this report is to discuss a study of the feasibility of implementing a model-based online management information system within the Illinois Library and Information Network. To that end, a prototype design is discussed. This discussion includes consideration of the projected operating costs of the management information system.

Several issues were considered in developing the prototype design presented here. While many of the issues are discussed in presenting the prototype design, the authors felt that these issues should be given fuller treatment. Thus, a substantial portion of this report reviews the general issues and how they are considered in the literature.

This work was supported by the Ilinois State Library under the Illinois Program for Title I of the Federal Library Services and Construction Act. The authors are especially grateful for the suggestions and encouragement of William DeJohn throughout this project.

Table of Contents


| | |
|---|---|
| Foreword | ii |
| Introduction | 1 |
| Characterizing Interlibrary Loan Networks | 5 |
| Mathematical/Statistical Issues | 5 |
| Information Processing Issues | 8 |
| Human Factors Issues - Data Entry | 10 |
| Human Factors Issues - Manager Usage of the MIS | 13 |
| A Prototype Design | 16 |
| MIS Control | 16 |
| Data Base Design | 19 |
| Data Entry | 19 |
| Manager-MIS Interaction | 27 |
| MIS Operating Costs | 33 |
| Conclusions | 36 |
| References | 37 |

# INTRODUCTION

A considerable number of mathematical models have been developed for use by library managers in analyzing their operations. Examples include book-use or circulation models [1,2,3], resource allocation models [4,5], and library network models [6,7]. However, not many models of library operations have become standard management tools. In other words, such models are infrequently used by library managers.

There seem to be two reasons for this lack of implementation. First, many models are of such sophistication that the typical library manager has difficulty understanding the assumptions and resulting equations. This problem can be substantially solved by developing a well-documented, interactive computer program that eliminates the need for the manager to deal directly with equations (e.g., see 8,9).

The second problem that arises when trying to implement library models is data collection. This problem is particularly constraining in situations where models should be frequently updated. Since data collection is often a rather labor-intensive activity, it can be quite expensive and it is difficult to justify the investment of resources in such a project if the only result will be data inputs for mathematical models. Thus, one must often find an operational need (e.g., acquisitions, cataloging, and circulation control) for the desired information if a data collection project is to be justified.

As an extreme example, one would have difficulty proving the necessity of an automated circulation system (e.g., CLSI, Gaylord, etc.) if its only purpose was to collect data for book-use models. Similarly, OCLC probably would never have been developed if its stated purpose had been to provide data on the use of their files as an interlibrary loan location tool. The circulation control and cataloging functions justified the developments of these systems. Nevertheless, both serve as potential sources of an abundance of data.

Since 1974, the authors have been developing a mathematical model for use in policy analyses of interlibrary loan networks. The model has been applied to three case studies [10,11,12] and a fourth study is underway. With the model and associated computer programs completed, most of the effort in performing a study involves data collection. Further, since policy analyses should be updated at least every year or so, data collection is a recurring problem. To justify such data collection efforts, one first has to recognize that the mathematical model is only one component in the network manager's decision making process.

Besides the forecasts of the model, the network manager needs a variety of reports summarizing current performance of the library network. In situations with which the authors are familiar, these reports are generated by a geographically disperse set of individuals, each of whom contributes a few entries to the final reports. The sum of all the individual efforts devoted to such data collection can be surprisingly

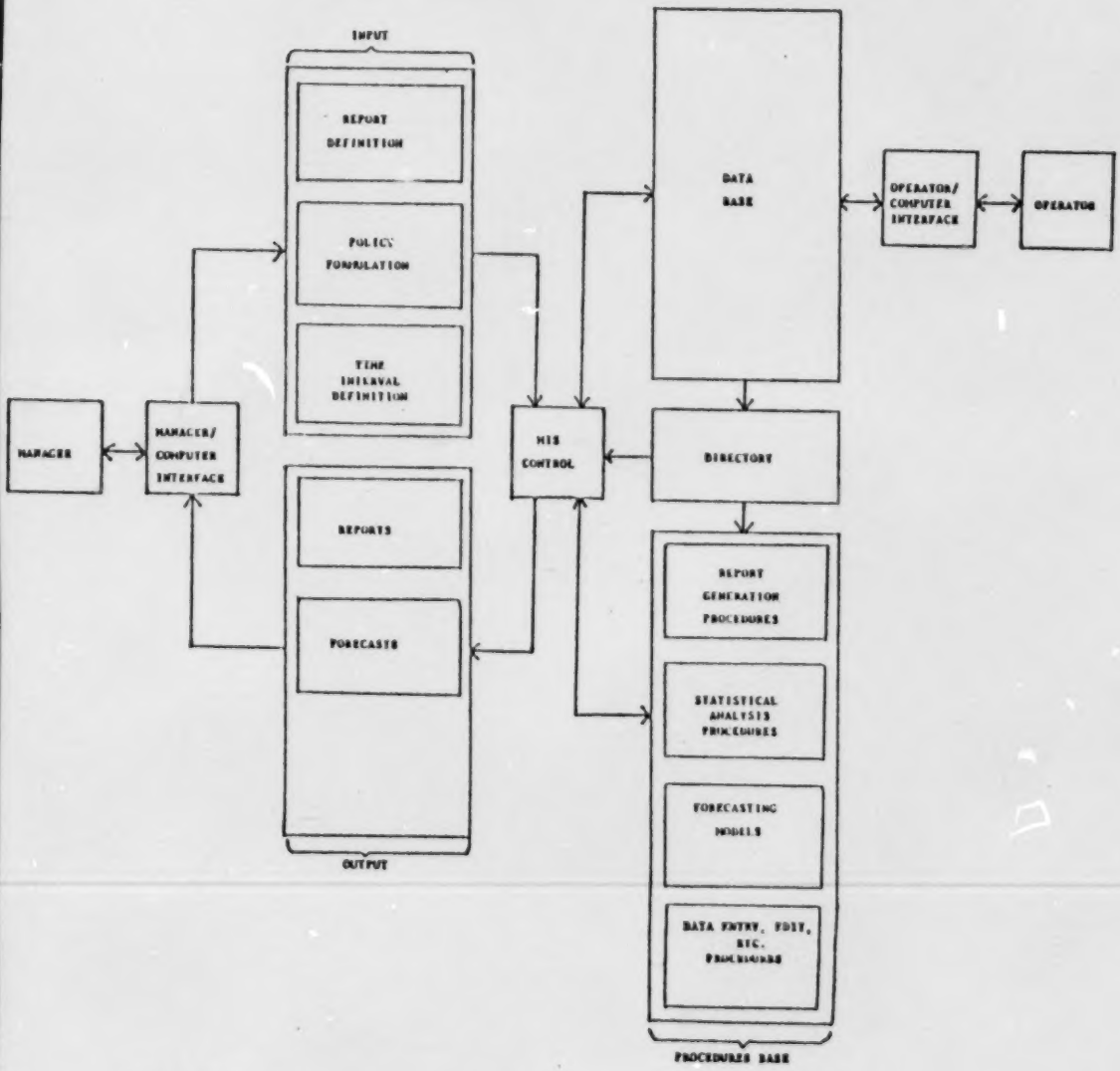
large. Thus, one finds that a substantial data collection project is continually underway and, the main reason this is not sensed by management is because the total effort is divided into so many small pieces.

This realization has led the authors to propose that data collection, report generation, and model-based policy analyses be centralized into an online management information system (MIS). Three advantages are foreseen. First, cost savings may be possible. However, this is difficult to predict since the filling out of reports is seldom a separate cost item in budgets. The second, and perhaps most important advantage, is the quality control possible with computerized data collection. Finally, the usual aggregations of raw data into simple overall statistics can be avoided and thereby allow many analyses prohibited with the highly aggregated data usually available.

The purpose of this report is to discuss the design of an online MIS for interlibrary loan networks. We will reference this discussion to the diagram in Figure 1. Our discussion will focus on three general areas:

1. Mathematical/statistical,
2. Information processing,
3. Human factors.

We will first consider the general issues in these three areas. Then, in the context of a prototype design of an online MIS for interlibrary loan networks, we will discuss how these issues have been specifically addressed.




Sketch of Management Information System Environment
Figure 1

## CHARACTERIZING INTERLIBRARY LOAN NETWORKS

To discuss our proposed MIS design, we first must characterize interlibrary loan networks. Such a network can be described in terms of requesting libraries, request classes, resource libraries, and request routing.*

A requesting library is any organization that generates interlibrary loan requests while a request class is any systematic division of requests into groups (e.g., subject class, requestor status, or mode of document delivery). A resource library is any organization that potentially might satisfy a request. Request routing refers to the chronological sequence of resource libraries where the request is processed as it seeks satisfaction.

An interlibrary loan network is further specified by the processing within resource libraries in terms of processing times, probabilities of success, and possibly processing costs. Finally, document delivery times (from resource library to requesting library) are needed to fully describe the network.

## MATHEMATICAL/STATISTICAL ISSUES

The reason for developing an MIS is to provide current status reports and forecasts to the manager. Thus, one plans to manipulate the data collected using various mathematical models and statistical procedures. The performance measures desired and the methodologies to be employed dictate the data to be



*As an individual library may be both a resource library and a requesting library one might instead refer to lending libraries and borrowing libraries, respectively. However, we will maintain our choice of words.

collected. Performance measures in this case have been defined as probabilities of success, turnaround times, and costs, and therefore the data collected must be appropriate for estimating the parameters necessary for the model that predicts these performance measures. The contents of the data base are thus a design specification and not open to much variation, (i.e., one cannot choose to utilize different data because it is easier to collect and process!)

Once the data is defined, the next issue concerns how much should be collected. In a library network that experiences many transactions, it may not be feasible to collect all of the data that is possibly available. Thus, some sampling scheme must be employed.

If a sampling procedure is adopted, one then must be concerned with how the status reports generated and model outputs will be affected. Succinctly, sampling leads to uncertainties that will impact the statistical quality of report entries and model predictions. This may preclude certain types of analyses and require higher levels of aggregation (i.e., pooling data across independent variables) than might be desired.

Time windowing (see Figure 1) is a feature which enables the MIS user to segment the data base by a given time interval. For example, assume an automated circulation system such as CLSI had been introduced into the network, and data regarding interlibrary loan processing before and after installation was available through the MIS. With a time windowing feature it might be

useful to focus on particular time periods with and without CLSI and compare status reports and forecasts. The selection of time periods will be affected by the sample size which may limit the minimum sizes of any time windowing. This will be due to inadequate sample sizes within small time periods.

Represented by the statistical analysis procedures block in Figure 1, are procedures which calculate the effect of input uncertainties on predicting network performance measures. Output from these procedures enables the manager to determine the statistical confidence of the model's predictions. More specifically, the procedures enable the manager to determine the number of request classes to use in policy analyses that will assure reasonably confident estimates of parameters to the model. Also related to sample size are statistical procedures which enable the manager to determine how much additional time is required to collect an adequate sample which will yield reasonably confident estimates of the parameters of the model.

The authors have discussed mathematical and statistical procedures elsewhere [6,11,13]. Other than noting that the forecasting models require certain data elements (i.e., demands, probabilities of success, processing times, and delivery times) we need not consider in any further detail the blocks labeled statistical analysis procedures and forecasting models.

Considering report generation, reports represent straightforward aggregations of data within and among requesting libraries, request classes, resource libraries, and the order of

resource libraries in request routes. In many cases, the desired reports are pre-defined, while in some situations, the MIS design process will include report design options to be specified by the manager. In this latter case, the ideas of Bariff and Lusk [14] may be helpful.

## INFORMATION PROCESSING ISSUES

While there are many design issues related to information processing within an MIS, two issues are particularly important: design of efficient execution procedures and design of the data base.

The design of efficient execution procedures concerns the MIS control block of Figure 1. Basically, one desires the various modules of the MIS to be easily available for execution without requiring an excessively large (and thus expensive) collection of programs to be simultaneously online. Also, the MIS should conveniently accomodate addition and deletion of modules. An obvious solution to this problem is to have the MIS user work at the "monitor" level of the operating system of the computer. However, this is probably unacceptable from a human factors point of view because communicating with the local operating system can be confusing to a user without much experience with computers. Further, the operating system is in general primarily concerned with protocols specific to the local computer facility.

An alternative solution to this problem is to design an interface between the user and operating system that accomplishes the task of executing different modules of the MIS by an overlay technique. Such an approach only requires that computer resources be sufficient for the largest module instead of the sum of all modules. This approach will be considered in more detail in the discussion of our prototype design.

A further alternative solution to the problem discussed here is the "virtual machine" idea of Donovan and Jacoby [15] which can be looked at as a generalization of the approach discussed in the previous paragraph. The virtual machine concept is more general in that it potentially allows users to simultaneously interact with multiple languages and operating systems.

Data base design involves developing efficient data storage and retrieval procedures. Efficiency can be measured with respect to the following objectives of information processing:

1. Minimize the number of disk accesses,
2. Minimize storage requirements,
3. Minimize response time or processing time,
4. Minimize vulnerability to system failure,
5. Accomodate access of data for multiple purposes [16].

Unfortunately, meeting all of these objectives is not always possible. For example, decreasing the number of disk accesses usually results in an increase in storage requirements.

A detailed discussion of the advantages and disadvantages of common file structures is beyond the scope of this report. Perhaps surprisingly, while various file organization methods are

available, there is no procedure which dictates the best type of organization for a particular application. One approach to avoiding this difficulty is to use a fairly general organization (e.g., see 17, Chapter 3) such as using inverted files for requesting libraries, request classes, resource libraries, routes, and dates of request initiation. This type of organization allows efficient answering of questions such as: How many requests from library A in the physics request class were satisfied by library B on May 23, 1977? However, this is not a realistic question since network managers can not possibly manage a network at such a low level.

The reports and forecasts needed by network managers are usually broken down by requesting library, request class, etc., and also aggregated across these dimensions. Thus, within a specified time interval (e.g., January-March 1977), the report generation or forecast generation modules will have to process all records. This has led us to conclude that the most effective data base organization for our purposes is sequential by request initiation date, with a small directory indicating where each date begins.

## HUMAN FACTORS ISSUES - DATA ENTRY

From Figure 1, one can see that both manager and data entry operator interact with the MIS. Within this section, we will consider the human factors of data entry. Most of the material discussed here, and in the next section, is condensed from references 18 through 20.

The operator's task is to process the information from a variety of forms where each form contains the record of the processing of an interlibrary loan transaction at a particular resource library. Multiple forms will be associated with a single request if it was processed at several resource libraries.

The goal in designing the operator-computer interface is to minimize data entry time while also minimizing errors. These are conflicting objectives in the sense that data entry operators usually make more errors when they attempt to increase their entry rate.

A well-designed format for the data entry display can improve entry rate. (An example of such a display and the specific considerations on which its design was based will be discussed later.) Entry rate can perhaps further be improved by using "defaults" that allow the human to avoid responding to queries by the computer when the response is obvious and/or irrevelant because of previous responses by the human. While this decreases the number of characters which the operator has to enter, it may be inappropriate where high entry rates are desired because the operator has difficulty programming himself to respond in a standard way when the set of computer queries varies depending on previous operator responses. For this reason, the design to be discussed later employs few defaults.

Even with a well-designed display, operators will make errors and the MIS designer should be prepared to deal with them. First of all, errors should not cause the execution of the

computer program to abort. For some systems, this may require that operator responses be read in a general alphanumeric format and then decoded into integers, etc.

The computer can be used to detect many errors. Unacceptable mnemonics, numbers that are too large or small, etc. can be easily checked and the operator informed of the error. If possible, and especially during learning, the computer should explain what is wrong in simple terms and suggest how the problem can be corrected.

More sophisticated error checking includes consistency checks (e.g., a request can not be received before it was initiated!). Another possibility is monitoring of the data entry rate (characters/min.) which can not exceed a certain value without errors being inevitable [21].

When the computer detects an error or, if the computer misses the error but the human catches it, some editing procedure is needed. Of course, one alternative is to simply start all over again. However, procedures can be designed to lessen editing time and also, make editing feel reasonably easy. Such a procedure will be illustrated during the discussion of our prototype design.

Regardless of how easy editing is, if a sufficient number of errors occur in entering a transaction, the operator may prefer to simply start all over again. Or, the operator may discover that he does not have sufficient information to enter a

particular transaction. In these cases, it is very useful to have an ESCAPE command that allows the operator to begin again.

## HUMAN FACTORS ISSUES - MANAGER USAGE OF THE MIS

At this point, we want to consider the human factors issues involved in the design choices related to manager-MIS interaction. The considerations are somewhat different than those for the operator because the manager is not in a forced-paced, high volume situation. Referring to the input and output blocks of Figure 1, one can see that the manager-MIS interaction is directly concerned with the forecasting models, report generation, and policy formulation procedures. We will consider the manager-MIS interaction in terms of two design issues: 1) how the manager understands the MIS, and 2) how the manager represents the problem with which he is interested.

To use the MIS effectively, the manager must understand how it works. This rather obvious statement can lead to designs where the vast majority of the software (perhaps 90%) supports manager-MIS interaction, as opposed to computation. While numerous dialogue structures are possible [18], the alternative chosen was a mixture of human-directed commands and computer-directed requests for the specifications of a command. Succinctly, commands designate procedures which, in turn, require specifications that may be several levels in depth. Inputs of the form of a simple "return" result in the dialogue moving "up" a level. Command errors or the typing of "HELP" lead to a

display of a specific reference to the User's Manual.

Related to the interactive structure of the MIS the manager should also understand the assumptions underlying computations and any defaults utilized. To this end, when possible, the system should check computations and assumptions during their use and indicate when assumptions have been violated. Although, of course, it is virtually impossible to check all assumptions.

For example, the manager has an option to run the model with or without the queueing analysis. Running the model with the queueing analysis implies that total request processing time is related to the average rate at which requests arrive. The queue utilization is defined as the portion of time a processing node is busy. It is the ratio of request arrival rate to service rate. When the value of the utilization exceeds 1.0, solutions to the queueing equations within the model will be incorrect and thus, the MIS user is warned that the predictions of network performance are not reliable.

Defaults are another consideration. Basically a default is when the computer infers what is supposed to happen rather than having the human actually specify it. If possible, whenever defaults are to be utilized, the manager should be aware of their usage.

Defaults employed by the operating system and/or compiler can be troublesome if the system is very forgiving. For example, some systems assume that "execution" commands that are not

followed by a program name refer to the last program which the user executed. This can be disasterous if, for example, the user last used a program for deletion of programs and then accidently types "execute" without specifying which program. Efforts should be made to program around this type of default.

On the other hand, some defaults can be particularly useful. For example, when the user responds with a simple RETURN rather than choosing one of the usual responses, the computer can infer that the user means "exit" or "no" or whatever is appropriate.

The second issue related to manager-MIS interaction is problem representation. Once the manager understands how the system works, he will then want to use it for policy analyses. A typical scenario will be for the manager to develop reports using the report generation procedures and then, based on what he gleans from these reports, to employ the mathematical models to forecast the implications of some new policy. This requires that the manager represent the policy of interest. Since a network usually includes many requesting libraries, request classes, resource libraries, etc., representing a policy can be complicated. The system to be explained below requires the manager to use a matrix-like format. Policy formulation can be somewhat simplified by forming new policies as edited versions of old policies. Yet, it is still a matrix-like format and one might intuitively think that a more graphic format would be more useful to the manager. However, it seems that large problems (many levels of numerous variables) such as library networks are

better expressed in matrix-like form [22].

## A PROTOTYPE DESIGN

At this point, we will discuss the specific ways in which the general issues discussed above have been addressed in the design of an MIS for the Illinois Library and Information Network (ILLINET). This discussion will include consideration of MIS control, data base design, data entry, manager-MIS interaction, and MIS operating costs. We will not consider the specifics of the mathematical/statistical issues since they are addressed elsewhere [6,11,13].

### MIS Control

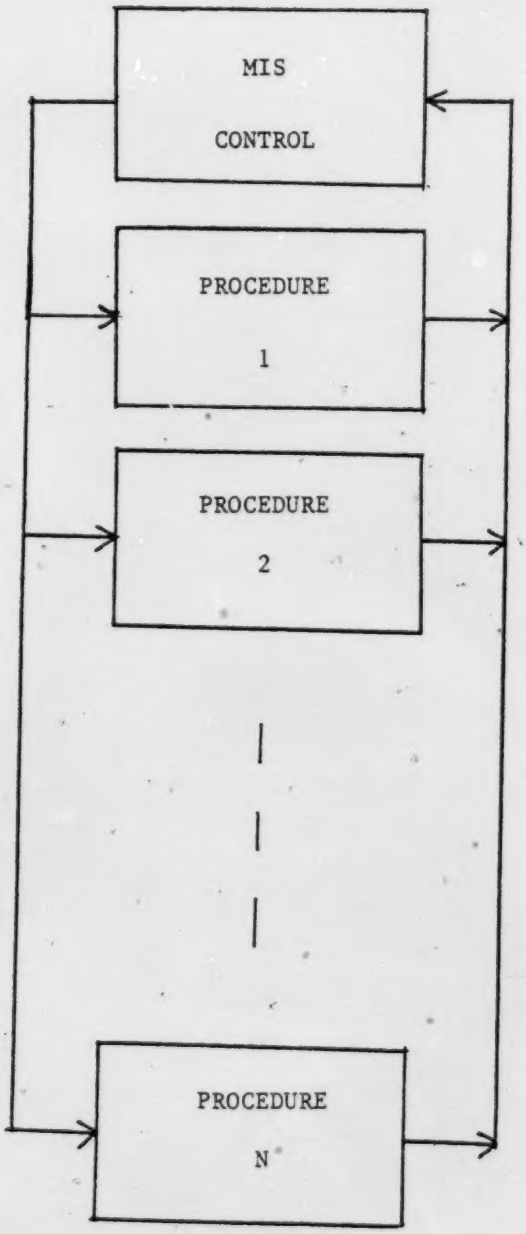
In Figure 1, there is a large block labeled procedures base. We have coined this phrase to represent a collection of programs which the user of the MIS invokes to perform various operations on the data base. The operator uses procedures such as ENTRY, UPDATE, and EDIT while the manager uses procedures such as REPORTS, MODEL, etc.

The choice of the MIS control structure for our prototype design was motivated by three objectives. First, it is very desirable to be able to add or delete procedures without affecting the overall MIS. Second, the user should have all procedures readily available without actually having them all online. Finally, the user should not be required to understand the "monitor" level of the computer system being used.

These considerations, especially the last one, led to the MIS control structure illustrated in Figure 2. Quite simply, each procedure is represented in the control structure by an entry in the procedures directory, depicted in Figure 1 between the data base and procedures base. If the user invokes a particular procedure, the MIS control first checks to see if the procedure exists. If the procedure is available, the MIS control executes the procedure by overlaying it in the same space used by the MIS control itself. This results in the MIS control ceasing execution while the procedure of interest is executing.

The last operation of every procedure is an instruction to execute the MIS control program by overlaying it in the space currently used by the procedure. Then, the MIS control is the only program executing. In this way a procedure can be added to the procedures base simply by adding this final instruction to the program.

The user initiates interaction with the MIS by typing RUN MIS which results in a * being displayed. He can then type ENTRY, UPDATE, MODEL, etc. and the appropriate program is executed. Upon completion of the procedure, the * returns. The user need not understand that MIS has actually ceased executing while, for example, ENTRY is executing. Further, the user never receives the numerous messages that are usual when one interacts with the computer at the "monitor" level.




MIS Control Structure
Figure 2

Data Base Design

As noted earlier, while there are many elaborate designs for data bases, it does not appear reasonable for us to adopt other than a simple structure. Succinctly, the design employed is to order the records in the data base chronologically by date of request initiation with a small directory to indicate where (i.e., what record) particular request initiation dates begin. This chronological arrangement is useful when, for example, reports are to be prepared for specific time intervals (e.g.,fiscal 1977). This aspect of MIS usage is represented in Figure 1 by the block labeled time interval definition.

Updating the data base is not performed directly. The operator enters new data into an update data base when executing the ENTRY procedure. This update data base is later merged with the regular data base by executing the UPDATE procedure. This approach maintains the integrity of the regular data base by assuring that all records in the update data base are computer-readable before they are merged with the regular data base.

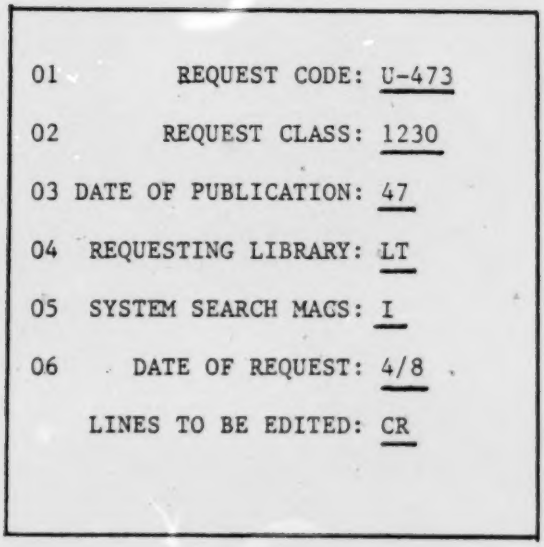
Data Entry

As discussed briefly above, the data entry operator enters data to the MIS from the various interlibrary loan forms sent to the network from the requesting libraries, while standardized teletype (TWX) forms are used by the resource libraries when routing requests to other network resource libraries.

Nevertheless, most of the data needed for the model and report generator are found in different formats and different locations on the currently used request forms. Thus, one of the functions of the MIS is to aid the data entry operator who must contend with this diversity of format and data on the interlibrary loan request forms.

The data regarding interlibrary loan processing naturally divides into two display types: one describing the initiation of a request and the other which relates to each of the resource library's processing of the request. Figure 3 illustrates the display generated by the MIS for data related to the identification and initiation of each request. Each data item of a request is identified by a sequential line number in the left margin of the display, which is useful to the operator for specifying changes due to input error. Figure 4 illustrates the data related to the resource library's processing of the request.

Some of the less-obvious entries on these displays require a little explanation. The request code (the first data item in Figure 3) serves to identify a unique transaction and link it to each referral or resource library processing the request. The first letter of the request ID code identifies the first resource library to receive the request. (Of course, a code with more than one letter would be necessary if there were more than 26 resource libraries.)


01 REQUEST CODE: U-473
02 REQUEST CLASS: 1230
03 DATE OF PUBLICATION: 47
04 REQUESTING LIBRARY: LT
05 SYSTEM SEARCH MAGS: I
06 DATE OF REQUEST: 4/8
LINES TO BE EDITED: CR

User responses are underlined.

CR indicates a simple return.

Data Entry Display: Identification of Request

Figure 3

```
                ROUTING OF REQUEST U-473

07                    RESOURCE LIBRARY 1: UOI
08                          DATE RECEIVED: 4/9
09 REC'D WITH MAC/CALL NO. (Y OR N): Y
10          INTERNAL SEARCH (Y OR N): Y
11   VERIFICATION REQUIRED (Y OR N): CR
12   ADDITIONAL INFO FOUND (Y OR N): CR
13                       OWNED (Y OR N): Y
14   OWNED BUT UNAVAILABLE (Y OR N): CR
15                    RESERVED (Y OR N): CR
16           PHOTOCOPY SENT (Y OR N): Y
17         LOCATION SEARCH (Y OR N): CR
18                           DATE SENT: 4/10
                    LINES TO BE EDITED: CR
```

User responses are underlined.

CR indicates a simple return.

Data Entry Display: Routing of Request

Figure 4

Many requesting libraries have access to a union catalog in the form of individual microfilm catalogs (MACs) for most resource libraries. When a requesting library notes that a given item was not found in a particular microfilm catalog, the identifying code of that particular resource library is entered (i.e., line number 5). For a given request not found in a particular MAC, one might say that the request was effectively sent to that resource library but, of course, did not require the usual processing tasks and turnaround time. This data can be used to measure the effect of previous referrals on probability of satisfying a request. In other words, is the probability of satisfying a request at a particular resource library related to the number of libraries which have been unable to satisfy the request?

Figure 4 illustrates the format for the data to be entered for each resource library. The first unnumbered line of the display is generated by the MIS from the previously entered request ID code data. Data for line 7 is also supplied by the MIS for the first resource library because by definition, the first resource library is identified by the first letter of the request ID code. Codes for identifying the second and subsequent resource libraries must be entered by the operator.

Data for lines 9 through 17 refer to the internal processing encountered by each resource library for each request. The first item, line 9, is derived from the original request forms and indicates whether the resource library is receiving the request

with a call number. Reference to the microfilm catalogs (i.e., MACs) accounts for those requesting libraries which send a photocopy of the catalog entry for the item they would like to borrow. Data for lines 10 through 17 are considered to be the significant (time-consuming) tasks encountered by the interlibrary loan staff of the resource libraries and include searching the card catalog or shelf list; verifying bibliographic information and whether or not that task resulted in additional information; whether or not the photocopy process was used; and whether or not an attempt was made to identify the location of the item in another library.

Now, we will discuss the specific issues considered in the design of the displays illustrated in Figures 3 and 4. An initial question was whether right or left justification of data entry would have an impact on length of time to enter data. Right justification of the MIS question and data input would require transmission of additional characters (i.e., leading blanks preceding the question). However, it was conjectured that the operator's task of checking the data and matching it with the appropriate question would be easier and thereby accomplished more quickly when input responses are lined up vertically. Thus, in the display design illustrated in Figures 3 and 4, right justification of all questions and data input was used.

As noted earlier, the issue of when to allow the computer to "interpret" particular responses or combinations of responses entered by the operator is an interesting issue. At one level of

interpretation, one can design the program to infer operator responses based on previous entries made by the operator. (In other words, "default" as discussed earlier.) For example, the request ID code not only identifies the unique transaction but also identifies the first resource library. Thus, the operator need not enter the latter piece of data. In this way, the display was designed to avoid entry of redundant data by the operator.

At another level of interpretation, one could also design a program which generates display forms tailored to each transaction based on the previous data. For example, the resource library form could be altered so that if the library did not engage in the verification process, there would be no need to ask about finding additional information. Or if a request were satisfied there would be no need to ask for location search activity. While the average number of lines per transaction would decrease with tailored displays, the average time might increase because the data entry operator might be more likely to read each question rather than routinely answer a standard set of questions. Thus, the data entry program standardizes the display forms for all requests.

Another consideration was use of the computer to check for errors related to acceptable alphanumeric format, fixed ID codes for libraries, and date consistency. All operator responses are read in alphanumeric format and decoded into the appropriate format. Error messages are generated when the decoded

alphanumeric entries are not recognizable. Incorrect identification of libraries is detected by the program and the operator is immediately supplied with a list of acceptable codes. Date consistency is checked for non-zero (i.e., known) dates by checking that the date sent by the requesting library is less than or equal to the date received by the first resource library. Similarly, other date checks are made within a resource library and between resource libraries for referred requests. For inconsistent dates, the operator is notified and a re-display of the appropriate form is presented with a pause for the operator to enter the correct and consistent dates. All related data for the particular form is re-displayed based on previous entry of data. Thus, the program forces the operator to correct errors related to ID codes, date consistency, and data format.

Once the data has passed the computer's data checks, the operator has the option of making any additional data changes before the transaction is stored on the disk. By entering the line number for the particular data entry on a given display, the operator will receive a re-display of that screen and data previously entered and be asked to supply the desired data. In the editing of data for lines 9 through 17 of Figure 4, the program will flip the current data ( either "yes" or "no") to the other value and re-display the updated input, thus avoiding a response from the operator other than a reference to the line number which should be changed.

If the operator comes to the conclusion that the number of items to be edited is too great, he can type ESCAPE which enables him to begin entry of the transaction all over again, or discard the current transaction and enter the next transaction.

In summary, the computer aids the operator by providing standard data entry forms, display of appropriate forms depending on the portion of the interlibrary loan transaction being entered, error checking for pre-coded and fixed format data, and checks for date consistency related to request processing times. The main responsibility of the human is to organize the data from diverse request forms and detect errors in the input which the program can not detect.

Manager-MIS Interaction

Figure 5 illustrates an example interaction of manager and MIS. The main purpose of this example is to illustrate how the manager can obtain information from the MIS to support his decision making process. Because we are focusing on the nature of this interaction, we have chosen a very simple problem that only involves a small set of network parameters.

We first need to describe the specifics of the hypothetical library networking problem. We assume that data entry is completed and the appropriate files are developed for input to the model (i.e., demands generated by requesting libraries, processing costs and times at resource libraries, etc.). We will assume that there are 20 requesting libraries, 2 resource

```
1       .RUN MIS
2       *MODEL
3       #MAKE
4       LABEL FOR NEW POLICY ? FIC TO RL1 AND NON TO RL2
5       RESOURCE LIBRARY ROUTE ? RL1,RL2
6       REQUESTING LIBRARIES USING ROUTE ? ALL
7       REQUEST CLASSES USING ROUTE ? FICTION
8       ROUTING IS NOT COMPLETE
9       RESOURCE LIBRARY ROUTE ? RL2,RL1
10      REQUESTING LIBRARIES USING ROUTE ? ALL
11      REQUEST CLASSES USING ROUTE ? NON-FICTION
12      #RUN
13      POLICY ? 1
14                REQUEST PROCESSING LOADS
15                     RESOURCE LIBRARY
16                   RL1      RL2      ALL
17      TOTAL       1500     1500     3000
18      SATISFIED    550      550     1100
19      UNSATISFIED  950      950     1900
20      SAT./TOT.   .367     .367     .367
21                PERFORMANCE PREDICTIONS
22                       REFERRAL
23                     0        1      ALL
24      FRAC.SAT.   .500     .050     .550
25      AVG. TIME   6.0      8.0      6.2
26      COST/SAT.   3.00     11.00    3.73
27      TOT.COST    3000     1100     4100
28      TABLE ? ALL
29      OUTPUT TO LINEPRINTER COMPLETE
30      POLICY ? _
31      #MAKE
32      LABEL FOR NEW POLICY ? ALL TO RESOURCE LIBRARY 1
33      RESOURCE LIBRARY ROUTE ? RL1,RL2
34      REQUESTING LIBRARIES USING ROUTE ? ALL
35      REQUEST CLASSES USING ROUTE ? ALL
36      #RUN
37      POLICY ? 2
38                REQUEST PROCESSING LOADS
39                     RESOURCE LIBRARY
40                   RL1      RL2      ALL
41      TOTAL       2000     1250     3250
42      SATISFIED    750      350     1100
43      UNSATISFIED 1250      900     2150
44      SAT./TOT.   .375     .280     .338
45                PERFORMANCE PREDICTIONS
46                       REFERRAL
47                     0        1      ALL
48      FRAC.SAT.   .375     .175     .550
49      AVG. TIME   4.0      10.0     5.9
50      COST/SAT.   3.67     4.57     3.95
51      TOT.COST    2750     1600     4350
52      TABLE ? ALL
53      OUTPUT TO LINEPRINTER COMPLETE
54      POLICY? _
55      #LIST
56      1 FIC TO RL1 AND NON TO RL2
57      2 ALL TO RESOURCE LIBRARY 1
58      #SVPOL
59      2 POLICIES STORED
60      # _
61      *EXIT
62      .
```

Manager - MIS Interaction

Figure 5

libraries, and 2 request classes, (i.e., fiction and non-fiction). The time period we have chosen for analysis assumes the initiation of 2000 unique interlibrary loan requests. We make a distinction between requests and transactions. We refer to unique requests as the number of requests initiated by the requesting libraries, and transactions as the total number of processings generated by the resource libraries in the referral route.

We also assume certain characteristics about the resource libraries in the network. Resource library 1 is strong in fiction and weak in non-fiction with probabilities of success in satisfying requests in these two subject classes of 0.50 and 0.25, respectively. Similarly, resource library 2 is weak in fiction but strong in non-fiction with 0.25 and 0.50 success probabilities respectively. All twenty requesting libraries are in the vicinity of resource library 1 with a one day average delivery time of requested items from this resource library to any requesting library. On the other hand, resource library 2 is fairly distant from requesting libraries and thus, average delivery time is five days.

The problem of concern to the manager in this example illustrates the tradeoff between turnaround times, probabilities of success, and costs. For this network, it appears obvious that fiction requests should first be sent to resource library 1, and non-fiction requests should first be sent to resource library 2. While the collection strengths of the resource libraries may

dictate this policy, the turnaround time of resource library 1 may dictate an alternative policy of first processing all request classes at resource library 1 (the 0th referral), and then referring the unsatisfied requests to resource library 2 (the 1st referral). The manager is faced with the decision of whether or not the time saved by processing non-fiction requests at resource library 1 is worth the extra cost due to 75% not being satisfied and thereby needing referral. Thus, the manager has defined the problem he must resolve. The MIS will provide him with predictions regarding probabilities of success, turnaround times, and costs. The final decision is based on a composite of factors, one of which is the predictions provided by the model.

All manager responses in this example are underlined and assumed to be followed by a "return". The line numbers to the left are for purposes of this discussion and do not actually appear on the display. After "logging in" and receiving the "monitor prompt" on line 1, the user types RUN MIS. The * appears on line 2 to indicate that he is within the MIS control structure. The user responds MODEL which, as noted earlier, results in the library network model procedure being laid over the MIS control structure and a # appearing.

On line 3, the user chooses to MAKE a policy which in this case involves routing requests first to the resource library most likely to satisfy them and then, if they are not satisfied, to the other resource library. On line 4, he specifies the label he has chosen for the policy. On line 5, he specifies the route.

On line 6, he could have specified each individual requesting library. However, it is more convenient to simply respond ALL. On line 7, he specifies that the RL1,RL2 route is for fiction requests. On line 8, the computer reminds him that a route is needed for non-fiction requests which he specifies on lines 9 through 11.

With the MAKE procedure complete, the # reappears whereupon he chooses the RUN procedure which asks him which policy he would like to run. He chooses policy 1. While the software could easily be designed to accept the label assigned to policy 1, experience has shown that the number of policies simultaneously under consideration is usually small and use of numbers expedites the interaction.

After choosing a policy, he immediately receives the forecasts of the library network model. This speed of response is possible because the performance forecasts are determined analytically and not via simulation which would require long run times to develop sufficient sample sizes and thereby could result in degraded human-computer problem solving performance.

Lines 14 through 20 show the predicted request processing loads throughout the network. The number of requests satisfied in this summary table include those satisfied on both the 0th and 1st referral. The probabilities of success on the 1st referral are usually lower than those on the 0th referral because of the fact that one resource library not being able to satisfy a request often indicates that another resource library is unlikely

to be able to satisfy it.

Lines 21 through 27 show performance predictions. The right column indicates that 55% will be satisfied with an average time from request initiation until item delivery of 6.2 days and an average cost of $3.73.

On line 28, the user is asked if he would like to see more detailed tabulations. By responding ALL, every tabulation available is output on the lineprinter including breakdowns by request classes, etc. On line 30, the user is asked if he wants to run another policy. By pressing the "return" key, he escapes to the higher # level.

At this point, the user wonders if he could reduce turnaround time substantially by avoiding the remote resource library 2 until after resource library 1 has been queried. While he could form this new policy by editing the previous policy (using EDPOL), it is simple enough to make a new policy which he does on lines 32 through 35.

On line 36, he runs the library network model and, on line 37, chooses policy 2. Lines 38 through 51 display the model's outputs. Comparing lines 25 and 49, one can see that average turnaround time is decreased by approximately 5%, while comparing lines 26 and 50 shows that the average cost of success is increased by approximately 6%.

On line 52, the user specifies that he wants a hardcopy of all tabulations. Then, on line 55, he checks to see what policies he has formulated. He decides to save these policies and types SVPOL on line 58. In this manner, he can access these policies (using RDPOL) when he next uses the model.

On line 60, he leaves MODEL and returns, on line 61, to the MIS control structure from which he chooses to EXIT.

There are many more procedures available to the manager within the prototype MIS discussed here. Within this section, we have not tried to be exhaustive. Instead, we have tried to illustrate the flavor of manager-MIS interaction.

MIS Operating Costs

There are several types of costs associated with operating an MIS. These include data preparation, entry, and storage as well as the costs incurred while the manager uses the MIS. We will address the former set of costs (i.e., those related to the data) and not consider the latter set associated with manager usage. This choice is dictated by the simple fact that, while we have considerable experience in using the manager-oriented procedures within the MIS, it is difficult if not impossible to specify exactly how a particular user will approach a particular problem.

On the other hand, we have used the prototype design discussed above (as well as an earlier version) during the course of two case studies where substantial amounts of data have been

collected, prepared for entry, entered, and stored. This process required two types of personnel: professional librarians and data entry operators. The librarians were necessary for assigning request classes (i.e., Dewey classification) to requests that were not satisfied. Further, they had to make decisions about the processing of requests where the information provided was ambiguous. The data entry operators had to sort the multiple forms, enter the data, and file the forms appropriately.

Based on the entry of data for 11,580 unique transactions, it was found that 11.92 hours per thousand transactions and 22.54 hours per thousand transactions were needed from librarians and operators, respectively. It is interesting to note that this implies a data entry rate of approximately 75 characters per minute for the operators. This is close to the rate found in an earlier study [12]. However, it is substantially below the rate found in a human factors handbook [23]. The reason is that the data entry operators have to pause during the entry process to make decisions about where the desired data is on the form. This leads one to realize that specially trained data entry operators will be needed, as opposed to traditional keypunch operators.

The computer costs associated with the above data entry effort included connect time, CPU costs, I/O costs, etc. on a DEC System-10. Data entry required 0.38 CPU hours, 4.60 kilo-core-hours, (a kilo-core-hour is the use of 1000 words of memory for one hour), and 15.58 connect hours per thousand transactions. For the DEC 10, this resulted in the total

computer costs of $86.33 per thousand transactions. Of course, for different computer systems where charges and computer speeds may vary, the cost would be different. However, this could be estimated using the CPU, kilo-core, and connect time numerics noted above.

The data base design employed in our prototype design resulted in requiring 26,000 words (36 bits/word) of storage per thousand transactions. The use of fixed length records made this figure somewhat larger than necessary. On the other hand, the use of a simple file structure helped to lessen storage and processing requirements. Since, the disk storage employed was rented for $30 per month, the costs of storage were insensitive to the amount of data collected.

To obtain a feeling for the implications of the total costs for operating the MIS, consider the total yearly processing load of approximately 150,000 unique transactions within the Resource Library level of ILLINET during fiscal 1977. This would require 1.75 operators and 1.0 librarians. Assuming $8,000 per operator and $12,000 per librarian to cover salary, benefits, and indirect costs, we obtain a yearly personnel budget of $26,000. Adding $12,949 for computer costs yields a total yearly budget of $38,949.

Of course, the MIS idea does not require that all possible data be collected. Sampling schemes have been developed for use with the library network models employed by the authors[13]. These schemes and the above cost estimates allow one to tradeoff

data collection costs and the resulting uncertainties in the reports and forecasts generated by the MIS.

## CONCLUSIONS

This report has considered the general issues in designing an online management information system for an interlibrary loan network environment. Further, a prototype design has been presented to illustrate the ways in which the general issues have been dealt with in a specific case.

At this point in time (November 1977), our prototype design is operating at the University of Illinois on a DEC System-10. The overall structure and many of the procedures are operational. Several data bases have been compiled. For several months, the design presented in this report has been used for data entry and policy analyses relating to several on-going case studies.

Beyond the specific MIS environment considered here, another purpose of this report has been to suggest that the success, in terms of implementation of mathematical models of library operations, requires some economical method of generating the data necessary to such models. A management information system seems to offer a vehicle for collecting data and implementing models.

Our experience in modeling library networks and in performing the work on which this report is based has led us to conclude that mathematical models can be successfully implemented. However, another issue has now emerged. Of what

value is the information provided by a management information system? Various controlled experiments have shown that management information systems improve the quality of management decisions [e.g.,24]. Yet, even though we know that information is an important ingredient in management decision making, we also know that in library networks [25], and in general [26], decisions are influenced by many factors of which information is only one. Thus, the question remains of whether or not a particular quantitative measure of worth can be assigned to management information.

## REFERENCES

1. Morse, P.M. 1968. Library Effectiveness, A Systems Approach. Cambridge: MIT Press: 1968.

2. Rouse, W.B. 1974. "Circulation Dynamics: A Planning Model." Journal of the American Society for Information Science. 1974 Nov/Dec; 25 (No.6): 358-363.

3. Chen, C.C. 1976. Applications of Operations Research Models to Libararies: A Case Study of the Use of Monographs in the Francis A. Countway Library of Medicine, Harvard University. Cambridge: MIT Press: 1976.

4. Kraft, D.H.; Hill, T.W. 1973. "A Journal Selection Model and Its Implications for a Library System." Information Storage and Retrieval, 1973 January; 9: 1-11.

5. Rouse, W.B. 1975. "Optimal Resource Allocation in Library Systems." Journal of the American Society for Information Science. 1975 May/June; 26 (No.3): 157-165.

6. Rouse, W.B. 1976. "A Library Network Model." Journal of the American Society for Information Science, 1976 March/April; 27 (No.2): 88-99.

7. Williams, J.G. 1976. "The Use of Simulation in Designing a Computer Based Library Network." Proceedings of the 39th Annual Meeting of the American Society for Information Science, Vol. 13, San Francisco, October 1976, 94.

8. Rouse, W.B.; Rouse, S.H. 1975. ILLINET: Interlibrary Loan and Information Network Model. User's Manual. Version 2. Coordinated Science Laboratory. University of Illinois at Urbana-Champaign, March 1975.

9. Rouse, W.B.; Rouse, S.H. 1975. "An Interactive Model for Analysis of Library Networks." Proceedings of the 38th Annual Meeting of the American Society for Information Science, Vol. 12, Boston, October 1975, 20-23.

10. Rouse, W.B.; Rouse, S.H. 1976. A Mathematical Model of the Illinois Interlibrary Loan Network. Project Report No. 3, Coordinated Science Laboratory Report No. T-26, University of Illinois at Urbana-Champaign, April 1976. ED124179.

11. Rouse, W. B.; Rouse, S. H. 1977. "Assessing the Impact of Computer Technology on the Performance of Interlibrary Loan Networks." Journal of the American Society for Information Science, 1977 March; 28 (No.2): 79-88.

12. Slate, M.P. 1977. Application of a Library Network Model: A Case Study of the Rolling Prairie Library System. Report No.6 in a Series Entitled A Mathematical Model of the Illinois Interlibrary Loan Network. Coordinated Science Laboratory Report No.T-40, University of Illinois at Urbana-Champaign, March 1977.

13. Rouse, W. B.; Rouse, S. H. 1976. "Data Collection and Parameter Estimation for a Library Network Model." Presented at the Annual Meeting of the Operations Research Society of America and The Institutes for Management Sciences, Miami, November 1976.

14. Bariff, M.L.; Lusk, E.J. 1977. "Cognitive and Personality Tests for the Design of Management Information Systems." Management Science, 1977 April; 23 (No.8): 820-829.

15. Donovan, J. J.; Jacoby, H. D. 1977. "Virtual Machine Communication for the Implementation of Decision Support Systems." IEEE Transactions on Software Engineering, 1977 September; SE-3 (No.5): 333-342.

16. Yourdon, E. 1972. Design of On-Line Computer Systems. New Jersey: Prentice Hall: 1972.

17. Lefkovitz, D. 1969. File Structures for On-Line Systems. New York: Spartan Books: 1969.

18. Martin, J. 1973. Design of Man-Computer Dialogues. Englewood Cliffs, NJ: Prentice-Hall, 1973.

19. Rouse, W.B. 1975. "Design of Man-Computer Interfaces for On-Line Interactive Systems." Proceedings of the IEEE, 1975 June; 63 (No.6): 847-857.

20. Rijnsdorp, J. E.; Rouse, W. B. 1977. "Design of Man-Machine Interfaces in Process Control." in H. R. Van Nauta Lemke and H. B. Verbruggen, Eds. Digital Computer Applications to Process Control. New York: North-Holland, 1977.

21. Neal, A. S. 1977. "Time Intervals between Keystrokes, Records, and Fields in Data Entry with Skilled Operators." Human Factors, 1977 April; 19 (No.2): 163-170.

22. Polich, J.M.; Schwartz, S.H. 1974. "The Effect of Problem Size on Representation in Deductive Problem Solving." Memory and Cognition, 1974 October; 2 (No.4): 683-686.

23. Van Cott, H.P.; Kinkade, R.G, editors. 1972. Human Engineering Guide to Equipment Design. Washington, D.C.: U.S. Government Printing Office, 1972.

24. Dickson, G. W.; Senn, J. A.;Chervany, N. L. 1977. "Research in Management Information Systems: The Minnesota Experiments." Management Science, 1977 May; 23 (No.9): 913-923.

25. Rouse, W. B. 1977. "Performance Criteria for Library Networks: Theoretical Basis and Realistic Perspectives." in A. Kent and T. Galvin, Eds. Library Resource Sharing. New York: Marcel-Dekker, 1976, Chapter 21.

26. Huber, G. P. 1977. "Predicting and Influencing Organizational Decisions." Systems Man, and Cybernetics Review, 1977 April; 6 (No.2): 2-5.